Homework

Class composition and cascaded function call

1. No. 10.9 in pp. 483, the textbook.

2. Implement a Simple Bookstore System

Objective:
Develop the `Book` and `Bookstore` classes to facilitate a simple bookstore system that supports operations like adding books, displaying books, and processing sales with cascaded function calls using the this pointer.

Provided Code:
You are provided with the `main()` function below. Your task is to implement the classes `Book` and `Bookstore` according to the functionality utilized in the `main()` function, ensuring methods support cascaded calls.

```
#include <iostream>
#include <string>

// Assuming MAX_BOOKS is a constant defining the maximum number of books
const int MAX_BOOKS = 5;

// You need to define the Book class with its members, constructor, copy
constructor, and methods
// You need to define the Bookstore class with its members, constructor, and
methods

int main() {
    Bookstore store;

    // Adding books with chained calls
    store.addBook(Book("1984", "George Orwell", 9.99))
```

```
        .addBook(Book("Animal Farm", "George Orwell", 7.99))
        .addBook(Book("The Great Gatsby", "F. Scott Fitzgerald", 14.99))
        .addBook(Book("A Brief History of Time", "Stephen Hawking", 15.99))
        .addBook(Book("Sapiens", "Yuval Noah Harari", 18.99));

    // Display all books and sell some books with chained calls
    store.displayBooks()
        .sellBook(1)  // Sell "Animal Farm"
        .sellBook(3); // Sell "A Brief History of Time"

    // Display total sales
    std::cout << "Total Books Sold: " << Book::totalSales << std::endl;

    return 0;
}
```

Requirements for Class Implementations:

- `Book` Class:
  - Attributes: `title`, `author`, `price`.
  - Include a copy constructor that logs when a book is copied.
  - Implement a method `display`() that prints book details and returns `*this`.
  - A method `sell`() that simulates selling the book, increments a static sales tracker, and returns `*this`.
- `Bookstore` Class:
  - Contains an array of `Book` objects and tracks the number of books.
  - `addBook(Book book)` method that adds a book to the array and returns `*this` for chaining.
  - `displayBooks()` method that prints details of all books in the store and returns `*this`.
  - `sellBook(int index)` method that sells a book by index and returns `*this`.